Міністерство освіти і науки України

Одеський національний політехнічний університет

Інститут комп’ютерних систем

Лабораторна робота № 6

По предмету «Програмування»

Тема: «Абстрактные и виртуальные методы. Методы класса. Переопределение операций»

Виконала: ст. групи АМ-173

Комлева Г. О.

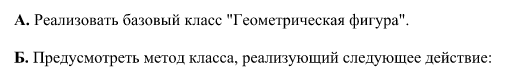
Перевірила: Головачова О. В.

Одеса 2018

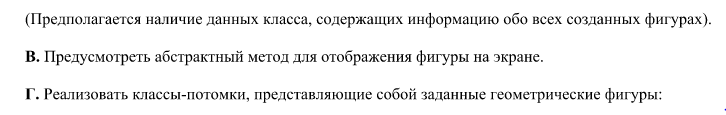
**Цель работы:**

Ознакомление и получение практических навыков при работе с абстрактными и виртуальными методами, используя методы класса и переопределение операций.

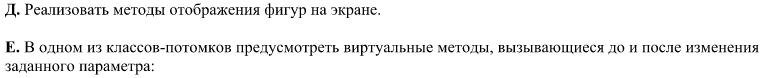
**Задание 2:**



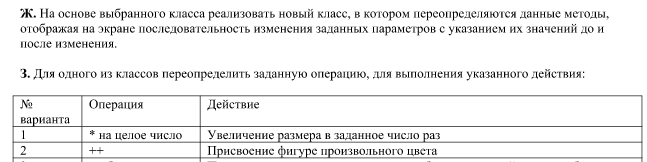












**Код программы:**

#include <iostream>

#include <string.h>

using namespace std;

static const int N=5;//количество фигур

class abs\_figures {

float coord\_x1[N];//координата x левой верхней точки фигуры

float coord\_y1[N];//координата y левой верхней точки фигуры

char color[20][N];//названия цветов фигур

public:

abs\_figures() {

for (int i=0; i<N; i++) {

coord\_x1[i] = 0;

coord\_y1[i] = 0;

color[i][0]='\0';

}

}

abs\_figures(float x[], float y[], char pc[][N]) {

for (int i=0; i<N; i++){

coord\_x1[i]=x[i];

coord\_y1[i]=y[i];

strcpy(color[i],pc[i]);

}

}

void change\_color() {

for (int i=0; i<N-1; i++)

strcpy(color[i],color[i+1]);

}

char\* get\_color(int i) {

if (i<0 || i>=N) {

cout<<"i is out of diapazone. Will be chosen figure # 1";

i=0;

}

return color[i];

}

float get\_x1(int i) {

if (i<0 || i>=N) {

cout<<"i is out of diapazone. Will be chosen figure # 1";

i=0;

}

return coord\_x1[i];

}

float get\_y1(int i) {

if (i<0 || i>=N) {

cout<<"i is out of diapazone. Will be chosen figure # 1";

i=0;

}

return coord\_y1[i];

}

void set\_color(char\* c, int i) {

if (i<0 || i>=N) {

cout<<"i is out of diapazone. Will be chosen figure # 1";

i=0;

}

strcpy(color[i], c);

}

void set\_x1(float x1, int i) {

if (i<0 || i>=N) {

cout<<"i is out of diapazone. Will be chosen figure # 1";

i=0;

}

coord\_x1[i]=x1;

}

void set\_y1(float y1, int i) {

if (i<0 || i>=N) {

cout<<"i is out of diapazone. Will be chosen figure # 1";

i=0;

}

coord\_y1[i]=y1;

}

virtual void show() = 0;

};

class arcs : public abs\_figures{ //класс наследуется от abs\_figures публично

char name[20][N];//название фигуры

float coord\_x2[N];//координата x конца дуги

float coord\_y2[N];//координата y конца дуги

float radius[N]; //радиус дуги

public:

arcs() : abs\_figures() {//вызывает конструкт родительского класса

for (int i=0; i<N; i++) {

coord\_x2[i] = 0;

coord\_y2[i] = 0;

name[i][0]='\0';

radius[i]=0;

}

}

arcs(float x1[], float y1[], float x2[], float y2[], char pn[][N], char pc[][N], float pr[]) : abs\_figures (x1, y1, pc){

for (int i=0; i<N; i++) {

coord\_x2[i] = x2[i];

coord\_y2[i] = y2[i];

strcpy(name[i], pn[i]);

radius[i]=pr[i];

}

}

void show() {

cout<<"Array of arcs\n";

for (int i=0; i<N; i++) {

cout<<"Arc # "<<i+1<<endl;

cout<<"Name - "<<name[i]<<endl;

cout<<"Color - "<<get\_color(i)<<endl;

cout<<"x1 - "<<get\_x1(i)<<endl;

cout<<"y1 - "<<get\_y1(i)<<endl;

cout<<"x2 - "<<coord\_x2[i]<<endl;

cout<<"y2 - "<<coord\_y2[i]<<endl;

cout<<"radius - "<<radius[i]<<endl;

}

}

char\* get\_name(int i) {

if (i<0 || i>=N) {

cout<<"i is out of diapazone. Will be chosen figure # 1";

i=0;

}

return name[i];

}

float get\_x2(int i) {

if (i<0 || i>=N) {

cout<<"i is out of diapazone. Will be chosen figure # 1";

i=0;

}

return coord\_x2[i];

}

float get\_y2(int i) {

if (i<0 || i>=N) {

cout<<"i is out of diapazone. Will be chosen figure # 1";

i=0;

}

return coord\_y2[i];

}

void set\_name(char\* n, int i) {

if (i<0 || i>=N) {

cout<<"i is out of diapazone. Will be chosen figure # 1";

i=0;

}

strcpy(name[i], n);

}

void set\_x2(float x2, int i) {

if (i<0 || i>=N) {

cout<<"i is out of diapazone. Will be chosen figure # 1";

i=0;

}

coord\_x2[i]=x2;

}

void set\_y2(float y2, int i) {

if (i<0 || i>=N) {

cout<<"i is out of diapazone. Will be chosen figure # 1";

i=0;

}

coord\_y2[i]=y2;

}

void set\_radius(float r, int i) {

if (i<0 || i>=N) {

cout<<"i is out of diapazone. Will be chosen figure # 1";

i=0;

}

radius[i]=r;

}

float get\_radius(int i) {

if (i<0 || i>=N) {

cout<<"i is out of diapazone. Will be chosen figure # 1";

i=0;

}

return radius[i];

}

};

class triangles : public abs\_figures{

char name[20][N];//название фигуры

float coord\_x2[N];//координата x второй точки треугольника

float coord\_y2[N];//координата y второй точки треугольника

float coord\_x3[N];//координата x третьей точки треугольника

float coord\_y3[N];//координата y третьей точки треугольника

public:

triangles() : abs\_figures() {//конструкт по умолчанию вызывает конструкт по умолчанию своего родителя

for (int i=0; i<N; i++) {

coord\_x2[i] = 0;

coord\_y2[i] = 0;

coord\_x3[i] = 0;

coord\_y3[i] = 0;

name[i][0]='\0';

}

}

triangles(float x1[], float y1[], float x2[], float y2[], float x3[], float y3[], char pn[][N], char pc[][N]) : abs\_figures (x1, y1, pc){

for (int i=0; i<N; i++) {

coord\_x2[i] = x2[i];

coord\_y2[i] = y2[i];

coord\_x3[i] = x3[i];

coord\_y3[i] = y3[i];

strcpy(name[i], pn[i]);

}

}

void show() {

cout<<"Array of triangles\n";

for (int i=0; i<N; i++) {

cout<<"Triangle # "<<i+1<<endl;

cout<<"Name - "<<name[i]<<endl;

cout<<"Color - "<<get\_color(i)<<endl;

cout<<"x1 - "<<get\_x1(i)<<endl;

cout<<"y1 - "<<get\_y1(i)<<endl;

cout<<"x2 - "<<coord\_x2[i]<<endl;

cout<<"y2 - "<<coord\_y2[i]<<endl;

cout<<"x3 - "<<coord\_x3[i]<<endl;

cout<<"y3 - "<<coord\_y3[i]<<endl;

}

}

triangles operator++() {//ереоределение операций

char str[20];

for (int i=0; i<N; i++)

set\_color(itoa(rand(), str, 10), i);

return \*this;//для объекта по ссылке this меняем цвет

}

};

class changed\_arcs: public arcs{

public:

changed\_arcs(float x1[], float y1[], float x2[], float y2[], char pn[][N], char pc[][N], float pr[]) : arcs(x1, y1, x2, y2, pn, pc, pr){}

void show() {

char f;

int num;

float qx1, qy1, qx2, qy2;

cout<<"Array of arcs before change\n";

arcs::show();//вызов метода show родительского класса дуг

do {

cout<<"Input the number of arc to change their coordinates (0 - "<<N-1<<"): ";

cin>>num;

if (num<0 || num>=N) {

cout<<"Number is out of diapazone. Will be chosen arc # 0\n";

num=0;

}

cout<<"Input x1 and y1: ";

cin>>qx1>>qy1;

cout<<"Input x2 and y2: ";

cin>>qx2>>qy2;

set\_x1(qx1, num);//используем сеттеры в явном виде

set\_y1(qy1, num);

set\_x2(qx2, num);

set\_y2(qy2, num);

cout<<"After change"<<endl;

arcs::show();//снова вызываем у род класса метод шоу

cout<<"Do you want to change coordinates? (Y/N)\n";

cin>>f;

} while (f=='Y' || f=='y');

}

};

int main() {

//Работа с дугами

float my\_coord\_x1[N];//координата x левой верхней точки фигуры

float my\_coord\_y1[N];//координата y левой верхней точки фигуры

char my\_color[20][N];//названия цветов фигур

char my\_name[20][N];//название фигуры

float my\_coord\_x2[N];//координата x конца дуги

float my\_coord\_y2[N];//координата y конца дуги

float my\_radius[N]; //радиус дуги

//зададим исходные данные для будущего перечня дуг

for (int i=0; i<N; i++){

cout<<"For ARC # "<<i+1<<endl;

cout<<"x1 = ";

cin>>my\_coord\_x1[i];

cout<<"y1 = ";

cin>>my\_coord\_y1[i];

cout<<"x2 = ";

cin>>my\_coord\_x2[i];

cout<<"y2 = ";

cin>>my\_coord\_y2[i];

cout<<"name = ";

cin>>my\_name[i];

cout<<"color = ";

cin>>my\_color[i];

cout<<"radius = ";

cin>>my\_radius[i];

}

//создаем ссылку на базовый абстрактный класс

abs\_figures\* f;

//создаем объект класса дуги, и теперь а-объект из массива пяти дуг

arcs a(my\_coord\_x1, my\_coord\_y1, my\_coord\_x2, my\_coord\_y2, my\_name, my\_color, my\_radius);

//устанавливаем ссылку на созданный объект

f = &a;

//теперь с ее помощью можно обращаться к методам объектa класса дуги

cout<<"\nOriginal arcs\n";

f->show();//метод шоу для arcs

f->change\_color();

cout<<"\nArcs after changed colors\n";

f->show();// поменялись цвета на послед

//создаем объект класса дуги с изменяющимися координанами

changed\_arcs b(my\_coord\_x1, my\_coord\_y1, my\_coord\_x2, my\_coord\_y2, my\_name, my\_color, my\_radius);

//устанавливаем ссылку на созданный объект

f = &b;

//теперь с ее помощью можно обращаться к методам объектa класса дуги с изменяющимися координанами

cout<<"\nWork with changed\_arcs\n";

f->show();//дуги с новыми коорд

//Работа с треугольниками

float my\_coord\_x3[N];//координата x третьей точки треугольника

float my\_coord\_y3[N];//координата y третьей точки треугольника

//зададим исходные данные для будущего перечня треугольников

for (int i=0; i<N; i++){

cout<<"For TRIANGLE # "<<i+1<<endl;

cout<<"x1 = ";

cin>>my\_coord\_x1[i];

cout<<"y1 = ";

cin>>my\_coord\_y1[i];

cout<<"x2 = ";

cin>>my\_coord\_x2[i];

cout<<"y2 = ";

cin>>my\_coord\_y2[i];

cout<<"x3 = ";

cin>>my\_coord\_x3[i];

cout<<"y3 = ";

cin>>my\_coord\_y3[i];

cout<<"name = ";

cin>>my\_name[i];

cout<<"color = ";

cin>>my\_color[i];

}

//создаем объект класса треугольники

triangles c(my\_coord\_x1, my\_coord\_y1, my\_coord\_x2, my\_coord\_y2, my\_coord\_x3, my\_coord\_y3, my\_name, my\_color);

//устанавливаем ссылку на созданный объект

f = &c;

//теперь с ее помощью можно обращаться к методам объектa класса треугольники

cout<<"\nOriginal triangles\n";

f->show();

++c;

cout<<"\nTriangles after change color\n";

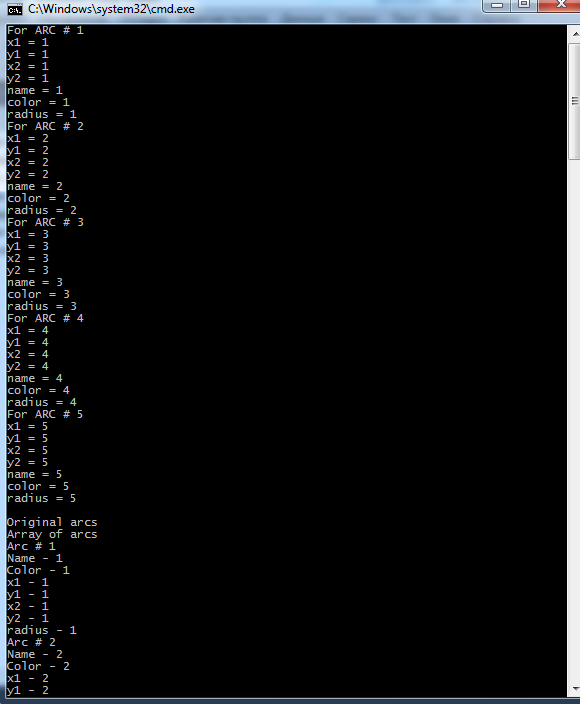
f->show();

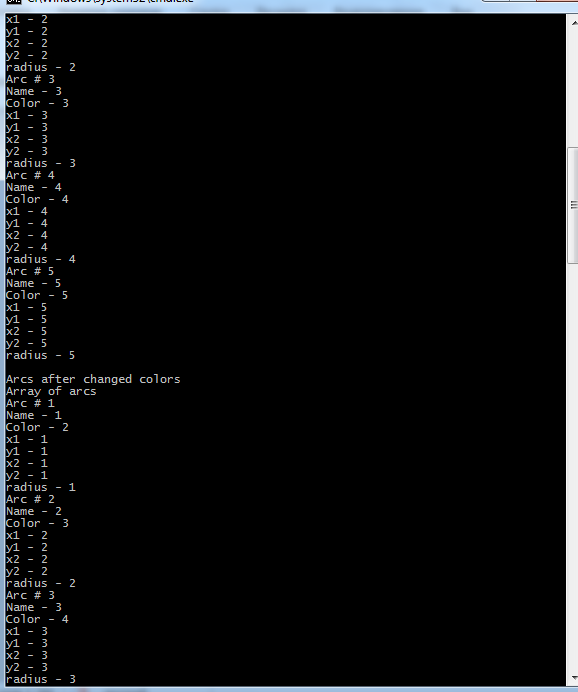
system("pause");

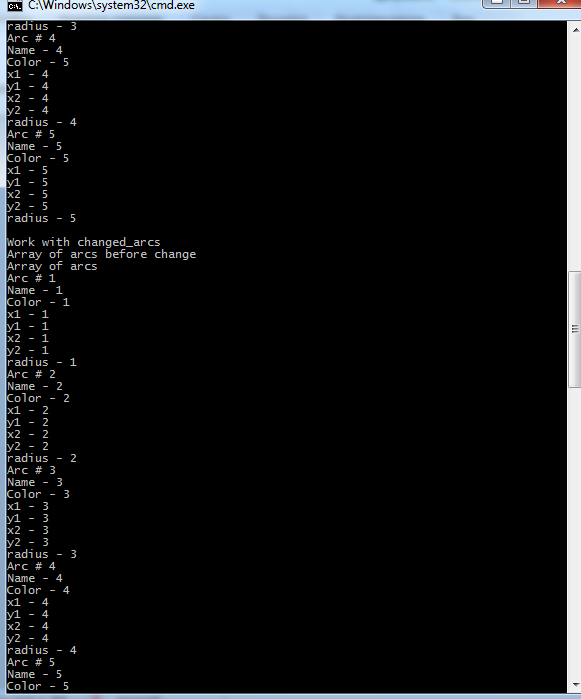
return 0;

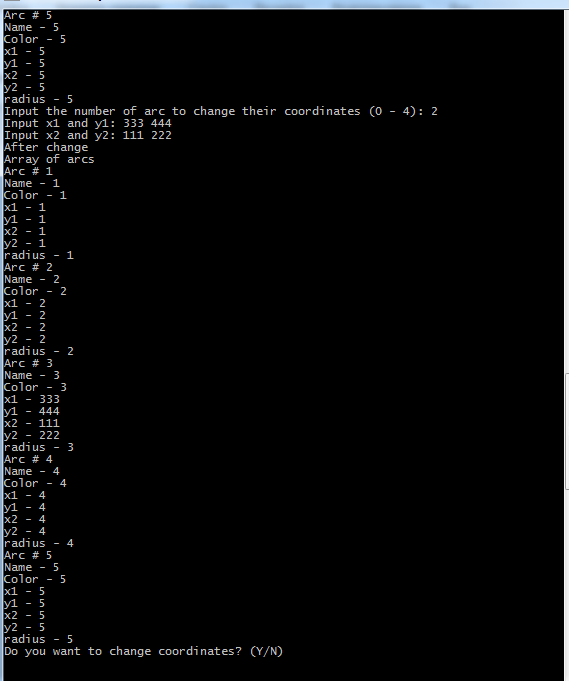
}

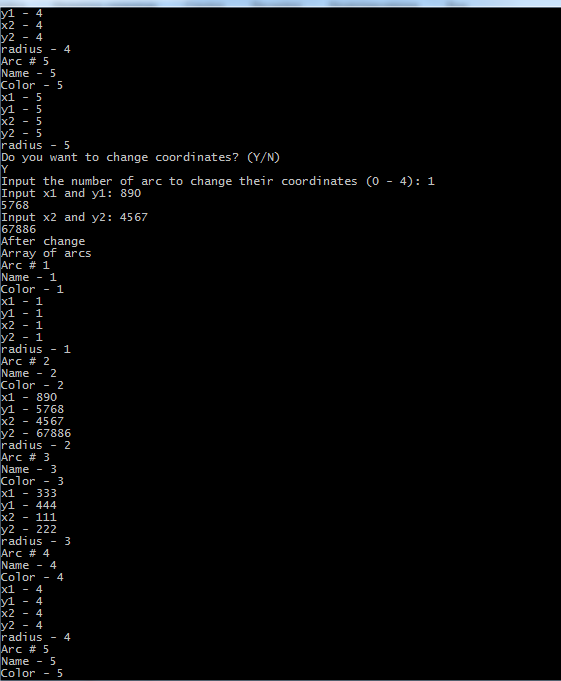
**Контрольный пример:**

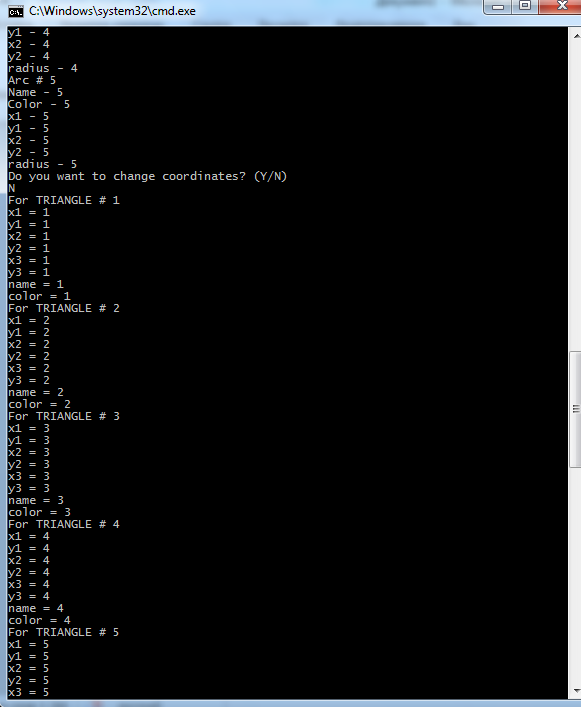


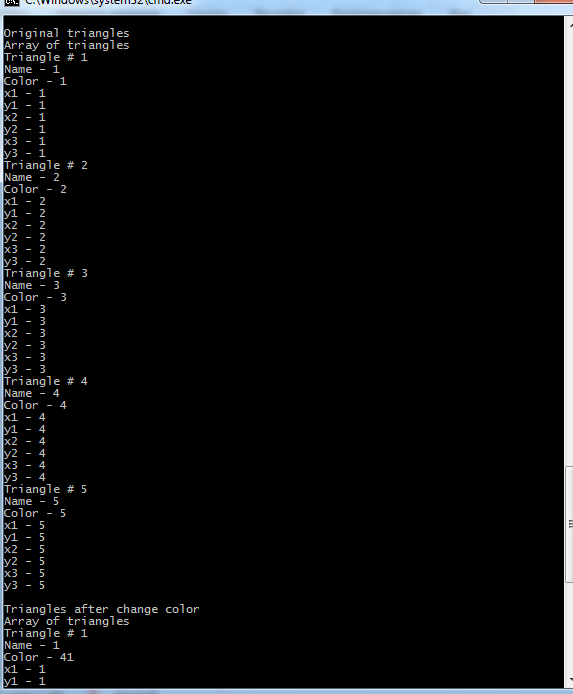


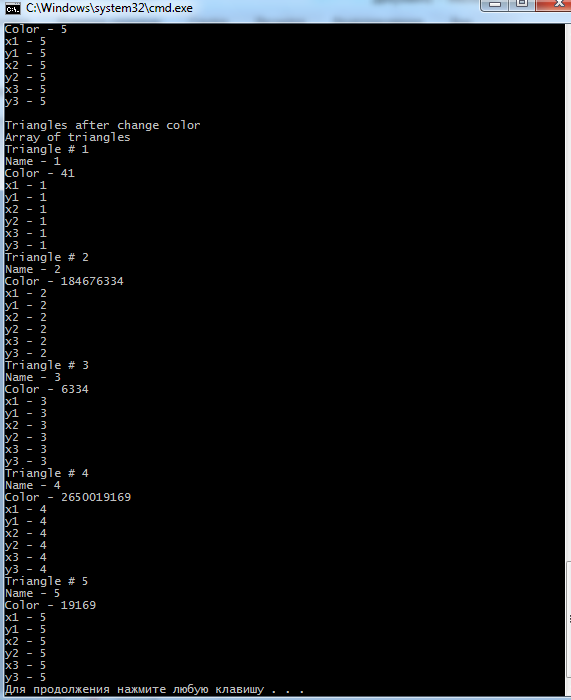












**Вывод:**

В ходе лабораторной работы я научиласьиспользовать абстрактные и виртуальные методы в программе и переопределение операций, используя при этом навыки работы с методами класса, с помощью чего была выполнена данная лабораторная работа.